REMARKS

Applicant respectfully requests reconsideration of this application. Claims 1-21 are pending. Claims 1, 3, 8, 10, 15, and 17 have been amended. Claims 2, 9, and 16 have been cancelled. No claims have been added. Therefore, claims 1, 3-8, 10-15, and 17-21 are now presented for examination.

Objection to Specification

The Advisory Action does not mention the previous objection to the lack of specification. The Applicant is proceeding with the understanding the previously presented arguments have been accepted. If the objection remains, the Applicant wishes to restate such arguments, and again submits that the application is presented in a form allowed by law.

Claim Rejections under 35 U.S.C. §102

Cheng, et al.

Archambault, et al.

The Examiner rejects claims 1-21 under 35 U.S.C. 102 (e) as being anticipated by U.S Patent Publication No. 2002/0010911 of Cheng, el al. ("Cheng").

The Examiner further rejects claims 1-21 under 35 U.S.C. 102 (e) as being anticipated by U.S Patent No. 6,173,444 of Archambault ("Archambault").

With regard to the previously submitted arguments, the Examiner has provided a reply indicating that the claims are not placed in condition for allowance. The Applicant wishes to respond to the contentions of the Examiner as stated below.

As amended herein, the provisions of claims 2, 9, and 16 have been merged into claims 1, 8, and 15 respectively, with clarifications being made, including clarification of a side-effect for the purpose of the claims. As amended herein, claim 1 reads as follows:

> 1). A method, comprising:
>
> analyzing each routine, of a software program having a plurality of separately compilable routines, to create a plurality of local side-effect lattice problems for each routine, a side-effect of a routine being the reading from or writing to a storage by the routine;
>
> merging the local side-effect lattice problems to create a global side-effect problem;
>
> computing a global solution to the global lattice problem; and
>
> splitting the global solution into local solutions for the local side effect lattice problems.

With regard to claims 1, 8, and 15, the Examiner has addressed the argument that Cheng and Archambault do not address the creation of local side-effect problems or the combination of such problems into a global side-effect problem. The Examiner states that the claims do not recite what the side-effect lattice problems and solutions are. The Examiner is directed to the application specification, which states, *inter alia*, that a side-effect may be a reading or writing of storage by a routine, and that a method may comprise determining for each pointer parameter of a routine whether that routine reads and/or write to storage via the parameter or a pointer derived from the parameter. (Application, ¶ 007) Claims 1, 8, and 15 have been amended to clarify that a side-effect of a routine is the reading from or writing to a storage by the routine. The application further indicates that a system and method compute a lattice value associated with each pointer parameter in a program, providing an example of a lattice in Figure 3.

The Examiner goes on to state that Cheng and Archambault disclose an element of a lattice domain such as the merge function that is the lattice meet operation of the greatest lower bound. To support this statement, the Examiner cites to Cheng and Archambault. Because the rejections are under the provisions of 35 USC §102 and thus each of the references is required to contain all the elements of the claims in order to support the rejection (any combination of the references being legally irrelevant under such provisions), the references are considered separately.

Cheng

For the first reference, the cited portion of Cheng provides as follows:

> [0106] Among many concrete values passed down from callers, only a small portion of them are necessary. For example, passing the address of a local variable as an actual parameter is insignificant unless the same address is passed through two different caller-accessible pointers, and both pointers are dereferenced by the callee. So in the phase III analysis, if the evaluation result of an access path found in the MOD/REF sections never has common right-most access paths with the evaluation results of other access paths, its bound values are excluded from the summary behavior since the memory access is always independent with other parameter dereferences across all calling contexts. *The trimmed summary behavior will be merged into each function to guide code optimizations.* The pseudo code listed in FIG. 12 summarizes various phases of analyses conducted in the interprocedural stage.

(Cheng, ¶ 0106, italics added for cited portion) In this portion, Cheng indicates that only certain values passed down from callers are necessary (with the example that the address of a local variable is insignificant unless the same address is passed through two different caller-accessible pointers, and both pointers are dereferenced by the callee). In this

regard, there may be an exclusion from summary behavior in certain circumstances. Cheng then states that the trimmed summary behavior will be merged into each function to guide code optimizations. Cheng thus generally indicates that summarized behavior is merged to guide code optimization.

The Examiner then indicates that the data-flow information is drawn from a lattic of possible data-flow information, cites to the following portion of Cheng:

> [0027] The third set (of pointer assignements) is considered as the summary transfer function and represented by points-to relations. In the interprocedural stage, bottom-up propagation of summary transfer functions along the call graph is performed. In the presence of function pointers, a top-down propagation of function names along the partially resolved call graph is also conducted, since some indirect call-sites may receive concrete function names through parameters. Because the transfer function of a just-discovered indirect callee may define function pointers used elsewhere in the program, *the bottom-up and top-down propagations need to be performed iteratively until a fixed point is reached. The aliases among formal parameters are then calculated after top-down propagation of concrete values along the complete call graph.* In the method of the invention, access paths enable a context-independent representation of transfer functions so that the memory overhead used to represent multiple versions of summary transfer functions is reduced.

(Cheng, ¶ 0106, italics added for cited portion) Is this portion, Cheng has described the propagation of functions, and indicated that bottom-up and top-down propagations are performed iteratively until a fixed point is reached. This iterative solution is not relevant to the claims, and clearly cannot apply to a process in which local problems are combined into a global problem for solution.

Cheng does not discuss the merger of local side-effect lattice problems to form a global side effect problem, the computation of a global solution to the global lattice problem, or the splitting of the global solution into local solutions for the local side effect lattice problems.

Archambault

With regard to the second reference, the Advisory Action cites to the following provisions:

> Conventional optimization techniques are generally intraprocedural; summary information for each procedure is collected during a first pass performed at compile time. The present invention also utilises interprocedural analysis, a second pass performed at link time, in which the collected information is merged and used to compute an interprocedural solution. The application is then re-built (re-compiled) using the interprocedural solution to optimize the application. A full discussion of a two-pass interprocedural analysis system can be found in Canadian Patent Application No. 2,102,089 titled "Recompilation of Computer Programs for Enhanced Optimization", which is commonly assigned and incorporated herein by reference. This application was laid open on Apr. 30, 1995.

(Archambault, col. 4, lines 19-33) Archambault thus discusses interprocedural analysis in general. However, as indicated in the application, conventional interprocedural analysis in a known technique. As indicated in Archambault, interprocedural analysis includes the used of a second pass performed at link time, in which the collected information is linked and used to compute the interprocedural solution. However, as clearly stated in Archambault, this is done at link time and thus, if a programmer edits a

unit, the program requires recompilation. The discussion of interprocedural analysis is a discussion of the known prior art, and does not address the elements of the claims.

The advisory action also cites to the following provision:

> The interprocedural analysis illustrated in FIG. 3 is essentially an extension of the intraprocedural analysis illustrated in FIGS. 1 and 2. In summary, all of the pointer graphs developed through the intraprocedural pass are gathered into a universal pointer alias graph for the whole program. Transitive closure is performed and the resulting reduced graph is used as input to a second interprocedural pass.
>
> Referring now to FIG. 3, each reduced pointer alias graph is read from memory (block 60). All definition nodes (from the accumulated graphs) for each pointer variable are merged into a single definition node, and the alias sets of each of the nodes are combined (union) to form the universal alias set for a specific pointer variable (block 62).

(Archambault, col. 7, lines 27-39) Once again, Archambault describes the prior art. The reference does not describe the creation of side-effect lattice problems and then combination of these side-effect lattice problems into a global side-effect lattice problem. Instead, the reference shows a conventional process of interprocedural analysis, involving an added second pass analysis of the program.

The Examiner states that "Cheng and Archambault disclose the lattice theoretic framework for the whole program interprocedural data-flow analysis, in the view of the broadest reasonable interpretation above." Again, it is emphasized that each of the references must be considered entirely separately under 35 USC § 102, and to support a rejection one of the references must contain all of the elements of the claims. As indicated above and as argued in the previous responses, neither of the references

contains all of the elements of the claims. The elements of the claims are not contained in either of the cited references.

Therefore, neither Cheng nor Archambault contains the elements of claim 1, and thus the rejections should be withdrawn.

It is submitted that the arguments presented herein also apply to independent claims 8 and 15. The remaining claims are dependent claims, and are allowable as being dependent on the allowable base claims.

Withdrawal of Appeal

Because Applicant has chosen more fully explain the arguments herein and to make clarifying amendments to the claims, the Applicant wishes to withdraw the current appeal to allow reconsideration of the claims.

Conclusion

Applicant respectfully submits that the rejections have been overcome by the amendment and remark, and that the claims as amended are now in condition for allowance. Accordingly, Applicant respectfully requests the rejections be withdrawn and the claims as amended be allowed.

Invitation for a Telephone Interview

The Examiner is requested to call the undersigned at (503) 439-8778 if there remains any issue with allowance of the case.

Request for an Extension of Time

The Applicant respectfully petitions for an extension of time to respond to the outstanding Office Action pursuant to 37 C.F.R. § 1.136(a) should one be necessary. Please charge our Deposit Account No. 02-2666 to cover the necessary fee under 37 C.F.R. § 1.17 for such an extension.

Charge our Deposit Account

Please charge any shortage to our Deposit Account No. 02-2666.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: 6/13/05

Mark C. Van Ness
Reg. No. 39,865

12400 Wilshire Boulevard
7th Floor
Los Angeles, California 90025-1026
(503) 439-8778